

# Методология подготовки больших данных для прогнозного анализа

Gubin E., Ph.D

•

# Информация об авторе

- **Evgeni Gubin**, Ph.D
- Associate Professor, TPU,
- Head of Master's Program "Big Data Solutions",
- Russia, 634050, Tomsk, Lenin Ave., 30,
- tel. 8(906) 958 7250,
- [gubine@tpu.ru](mailto:gubine@tpu.ru)

# Видео уроки по SAS

---

<https://habr.com/ru/company/sas/blog/348168/> -SAS уроки

<https://drive.google.com/file/d/1j56iO9cRZL4OehOM8MqWpuPbxOdxWryF/view>

Хорошая книга по SASy

<https://video.sas.com/detail/videos/how-to-tutorials> - обучение SAS Base!!!

- Credit risk modeling. Design and Application. Elizabeth Mays, Editor. Glenlake Publishing Company Ltd. 1998
- Introduction to Scorecard for Model Builder. Fair Isaac Corporation. 2008 – 40pp.
- Applying Data Mining Techniques Using Enterprise Miner Course Notes/ SAS Institute Inc. Cary, NC 27513, USA, 2003
- The Little SAS Book: A Primer Second Edition. Cary, NC: SAS Institute

## [Программирование на Python · Stepik](#)

- <https://stepik.org/course/67/promo>
- <https://stepik.org/course/512/promo>
- [ОСНОВЫ СТАТИСТИКИ. Часть 2 · Stepik](#)
- <https://stepik.org/course/76/promo>
- <https://stepik.org/course/524/promo>

**Цель:** формирование базовых компетенций слушателей при подготовке исходных данных в области больших объемов данных (**На примере использования технологий SAS**).

---

- **О курсе:** Процесс сбора и подготовки исходных данных, является одним из самых трудоемких и сложных этапов в анализе больших объемов данных, который порой занимает до 80% всего времени. Использование статистических методик и современного программного обеспечения позволяет значительно сократить временные и финансовые затраты на данном этапе и повысить эффективность и качество конечных результатов.
- **Аудитория:** Архитекторы Data lake, Аналитики данных, Дата-инженеры, отвечающие за процессы сбора, подготовки и очистки данных.
- **Предварительные требования:** базовые знания в области программирования, высшей математики, статистики.

## 1. Введение в Data Mining

- Процессный подход **Data Mining. Data Lake** концепция. Стандарты **CRISP-DM** и **SEMMA**.
- Фазы жизненного цикла процессов **Data Mining**.
- Подход **Data provenance** - происхождение данных.
- Подход **Data Lineage** и документирование.

## 2. *Описательная статистика.*

- Бизнес-анализ – начальная фаза, на которой происходит определение бизнес-целей и вырабатываются требования к результатам. Происходит первичный анализ данных, который включает проверку качества данных и простейшие статистики, исправление ошибочных и противоречивых данных. В данном разделе предполагается оценить форматы входных параметров (**объясняющих переменных**) и сформулировать **целевую функцию**.
- Кроме этого проводят интерактивный статистический анализ отклонений с помощью визуализации данных.

### 3. Подготовка данных

- Подготовка данных включает очищение данных (анализ пропущенных значений и выбросов (Outliers)), удаление дублирующих строк. Важной составляющей подготовки данных является выявление мультиколлениарности в объясняющих переменных и ее наличие позволяет удалять эти переменные. Нормализация (преобразование, масштабирование) позволяет преобразовать исходные данные в единый формат, что избавит от влияния разноформатности объясняющих переменных на целевую функцию.

### 4. Разбиение исходных данных

- Формирование тренировочной и тестовой выборки, их содержание и объем по отношению ко всему объему исходных данных обычно составляет 70:30 и обязательное требование репрезентативности. Основные принципы формирования тестовой выборки. Если используется несколько обучающихся моделей и данных достаточно много, то добавляют валидационную выборку в соотношении 60:20:20.

# Программа курса (продолжение)

---

## 5. Модели для анализа данных.

- Понятие регрессионных моделей и их сильные и слабые стороны. Деревья решений и их особенности.

## 6. Заключительный проект.

- Выполнение полного цикла очистки и подготовки данных на примере выбранного `data set`.
- Формирование обучающейся и тестовой выборки. Документирование.



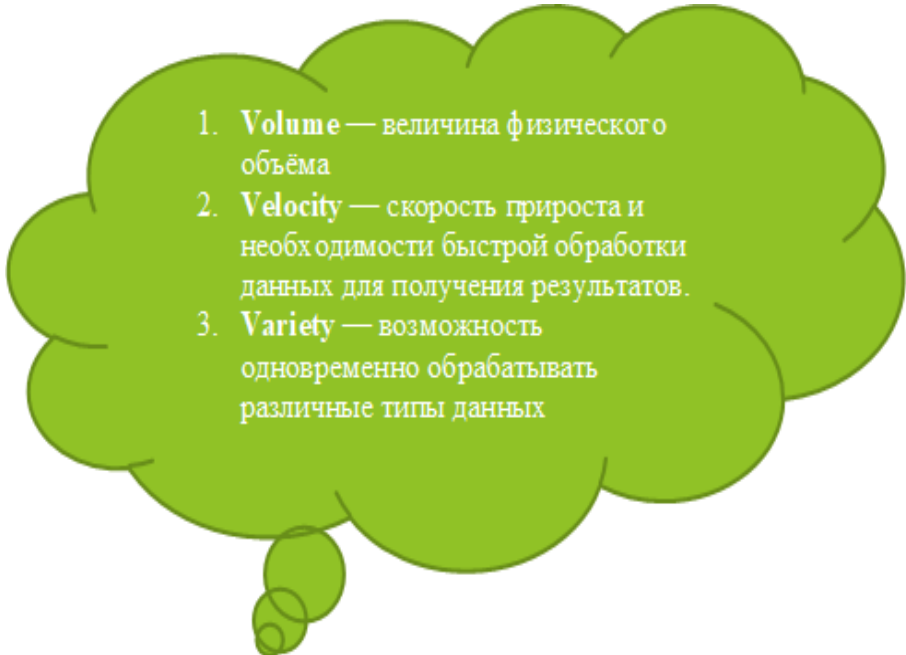
## Цель: изучение современных технологий для подготовки больших данных

---

Процесс сбора и подготовки исходных данных, является одним из самых трудоемких и сложных этапов в анализе больших объемов данных, который порой занимает до 80% всего времени. Использование статистических методик и современного программного обеспечения позволяет значительно сократить временные и финансовые затраты на данном этапе и повысить эффективность и качество конечных результатов.

**Big Data** – это *горизонтально масштабируемая система*, использующая набор методик и технологий, позволяющих обрабатывать *структурированную и неструктурированную информацию и строить связи*, необходимые для получения однозначно интерпретируемых человеком данных, не успевших потерять актуальность, и несущая ценность преследуемых им целей.

# Big Data – три источника и три составных части - «V»

- 
1. **Volume** — величина физического объёма
  2. **Velocity** — скорость прироста и необходимости быстрой обработки данных для получения результатов.
  3. **Variety** — возможность одновременно обрабатывать различные типы данных

## Big Data

*Internet*(social networks, forums, blogs, media and other sites)

**Corporate archives of documents**

*Readings of sensors, instruments and other devices*

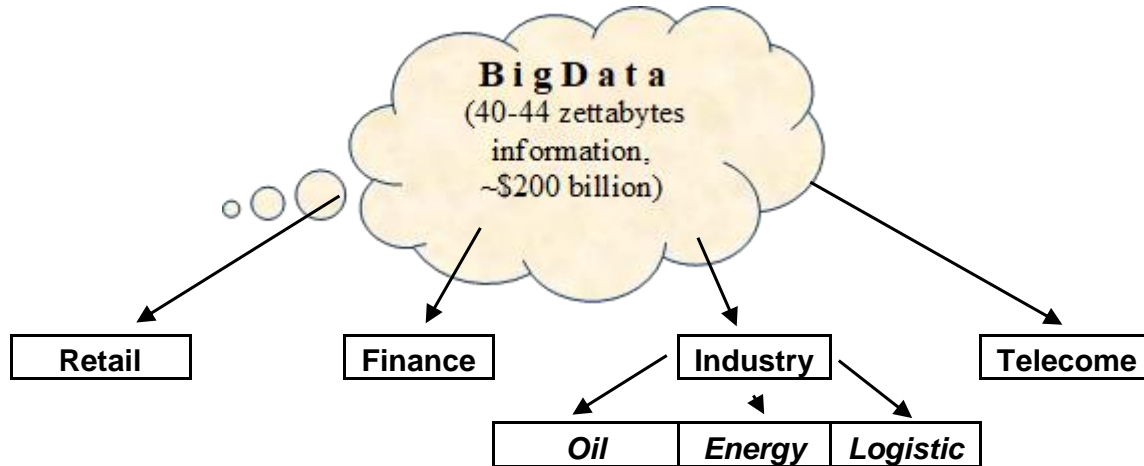
## Big Data – Три «V»

---

- Во-первых, не всегда большой объем данных говорит о системе, что она решает вопросы Больших Данных.
- Во-вторых, важно поддерживать необходимую *скорость обработки* поступающих данных, иначе можно потерять их ценность и передать на дальнейший анализ уже не валидные данные или в качестве результата предоставить неактуальную информацию.
- В-третьих, важно уметь находить *связи между любыми данными*, вне зависимости от уровня их структурированности, и уметь получать результат, который можно однозначно анализировать для решения той или иной задачи.
- В-четвертых, система должна быть хорошо *масштабируемой* на уровне логики, иначе мы рискуем получить недостоверные данные ввиду потери одного из магических V, потери которого неизбежны при наличии большего потока информации, нежели мы можем обработать.

# Big Data – кто работает с большими данными

- **Big Data** – это горизонтально масштабируемая система, использующая набор методик и технологий, позволяющих обрабатывать структурированную и неструктурированную информацию и строить связи, необходимые для получения однозначно интерпретируемых человеком данных, не успевших потерять актуальность, и несущая ценность преследуемых им целей



# Крупнейшие компании работают с большими данными

## 1. Retail

- Walmart (486 млрд.\$), Costco (119 млрд.\$), The Kroger(115 млрд.\$) – US, X5 Retail Group (1287 млрд. P), «Магнит» (1287 млрд. P), «Лента»(365 млрд. P)

## 2. Finance

- HSBC (GB, 2634 млрд. \$), BNP Paribas (Франция, 2514 млрд. \$),
- Sberbank (24 680 млрд. P), ВТБ (24 680 млрд. P), Газпромбанк (5 742 млрд. P)

## 3. Industry

- Exxon Mobil(US), Royal Dutch Shell (GB), British Petroleum(GB), General Electric(US)
- Gazprom, Resent, Lukoil, Russia's railways

## 4. Telecom

- Huawei (China), Google (US), AT&T(US), Sprint (US), T-Mobile US, Facebook
- МТС, Мегафон, Yandex, «WhatsApp», «Одноклассники»

# Источники больших данных

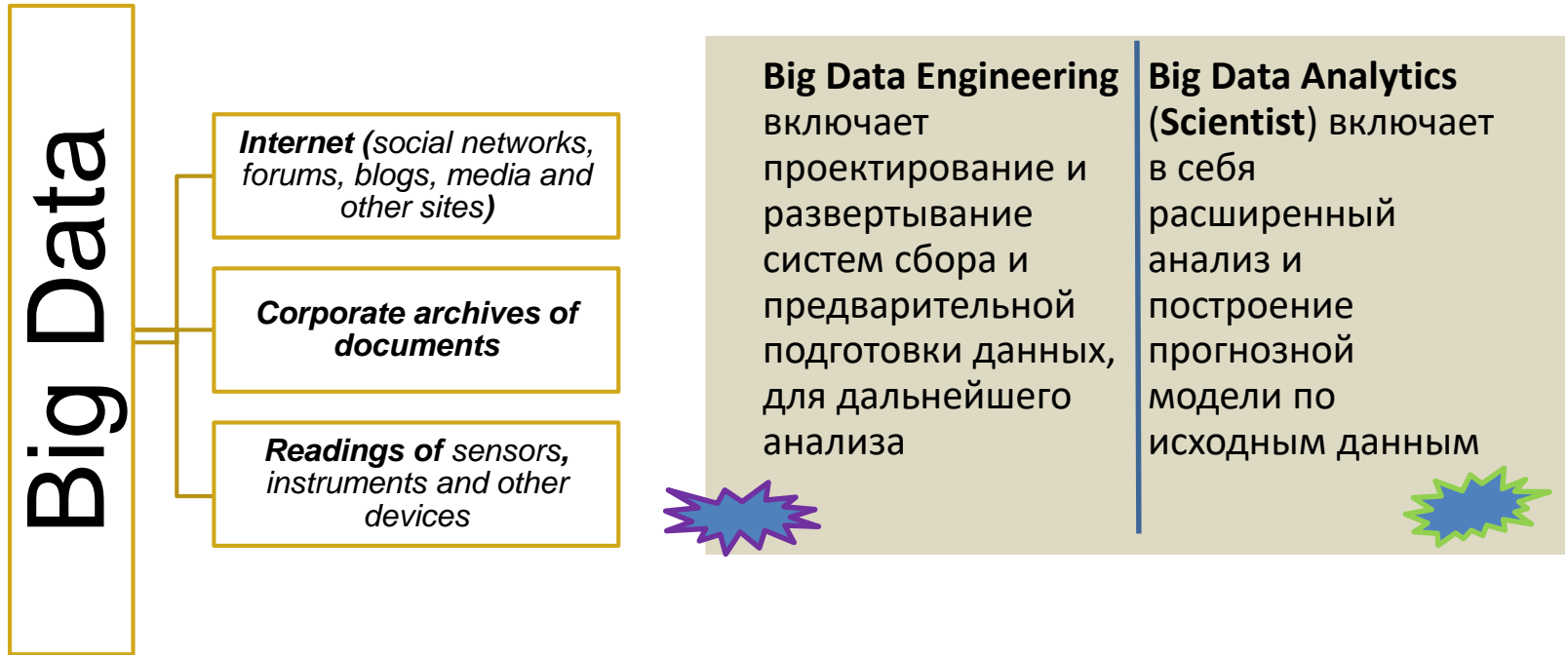
- Структурированные данные

- ✓ ETL – хранилище данных
- ✓ Таблицы
- ✓ Файлы
- ✓ Внутренние источники данных (**CRM**, **ERP**, и др.)
- ✓ Внешние источники данных (соцсети, интернет, и др.)

- Неструктурированные данные

- ✓ Файлы без predetermined модели
- ✓ «**Озера данных**» -«Идея озера данных состоит в том, чтобы хранить необработанные данные в их оригинальном формате до тех пор, пока они не понадобятся».

# Big Data – Технологии и основные источники поступления больших данных



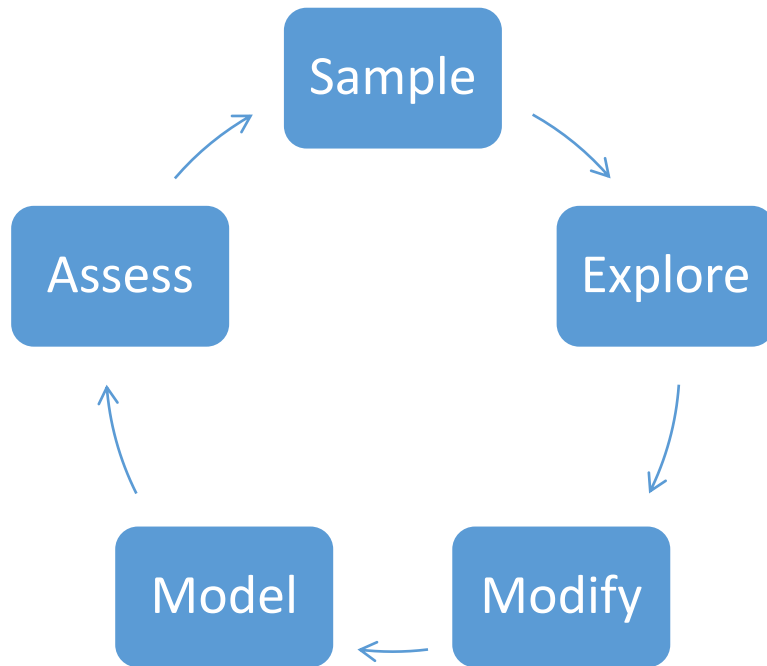
- В настоящее время Data Mining («интеллектуальный анализ данных») определяется как процесс поиска скрытых закономерностей в больших объемах данных, которые часто не структурированы и имеют разнообразные форматы (в виде чисел, текста, фото и т.п.).
- Большую часть времени интеллектуального анализа данных приходится тратить на подготовку данных: очистку, агрегирование, преобразование и моделирование. Другой проблемой является то, что статистические модели часто строятся на данных с большим количеством наблюдений или переменных. Для масштабируемости необходимо тщательно выбирать и применять статистические методы.
- Институт SAS определяет интеллектуальный анализ данных *как процесс выборки, изучения, изменения, моделирования и оценки (SEMMA) больших объемов данных для выявления ранее неизвестных шаблонов, которые могут быть использованы в качестве бизнес-преимущества.*



- SAS Enterprise Miner был разработан для поддержки всего процесса интеллектуального анализа данных. Система SAS обеспечивает доступ к реляционным и гетерогенным хранилищам данных. Базовый язык SAS обеспечивает высокую мощность в агрегировании и преобразовании данных. Вместе SAS/STAT и Enterprise Miner могут поддерживать высокую практическую реализацию в численном моделировании исследуемого бизнес-процесса. Функции Enterprise Miner организованы в известный логический алгоритм как **SEMMA**:
  - **SAMPLE**-определение входных данных, в том числе выборку (sample) из больших данных
  - **EXPLORE**-исследование набора данных статистически и графически
  - **MODIFY**- подготовка данных для анализа
  - **MODEL**- оценка предсказательной модели (регрессионная модель, деревья решений, нейронные сети и др.)
  - **ASSESS**- сравнение конкурирующих предсказательных моделей (графическое сравнение исследуемых респондентов, графики доходности и т.п.)

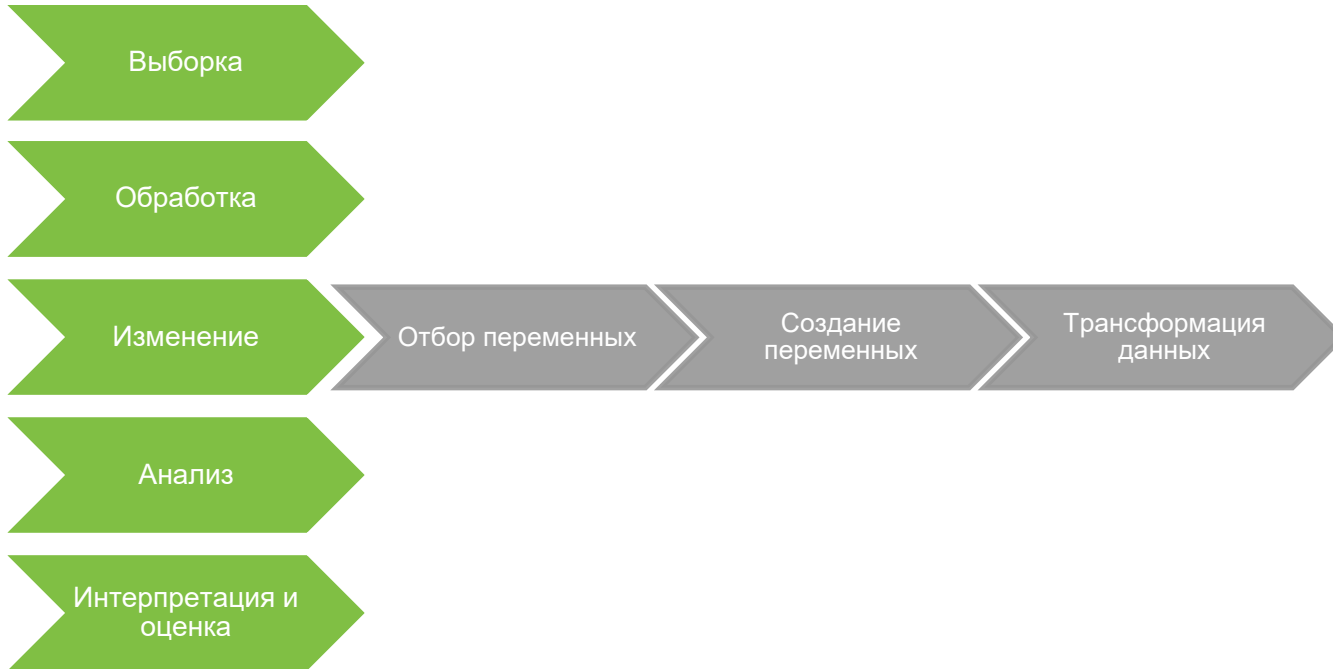
# SEMMA - структура

---

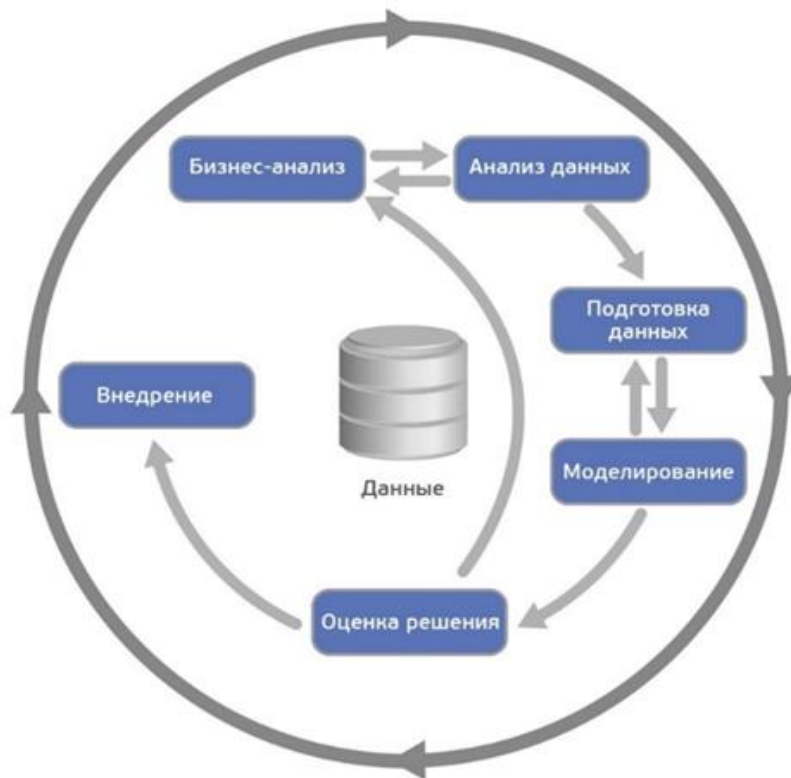


# Обзор существующих подходов. SEMMA

---



# Методология CRISP-DM

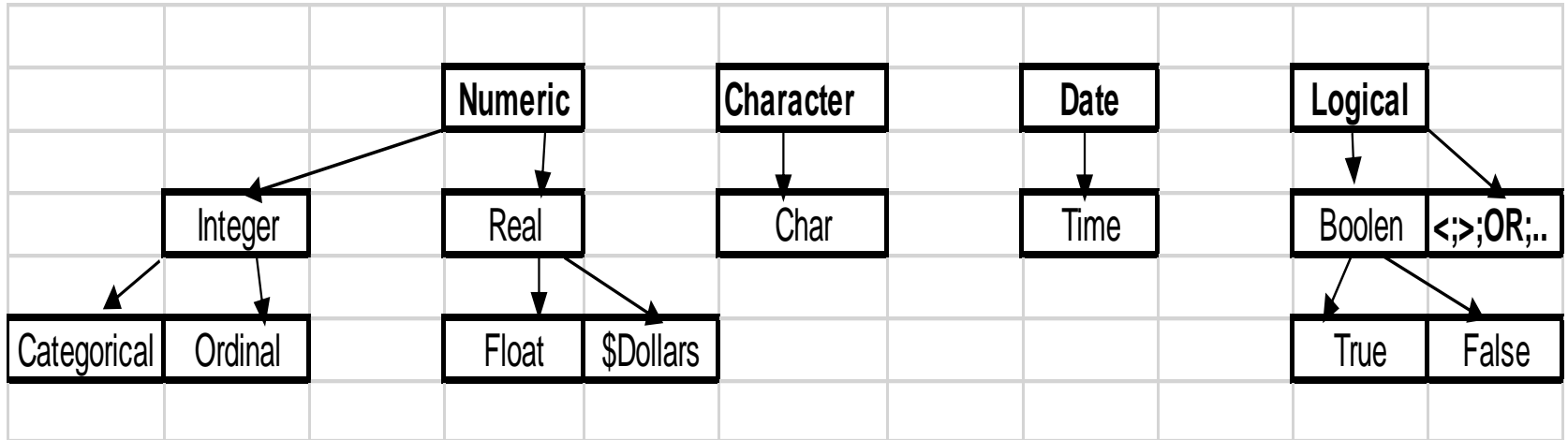


# Обзор существующих подходов. CRISP-DM

---



# Форматы данных



## Подготовка данных

ID	Sex	DocType	Business	Education	CarStatus	AddIncome	Income	CreditSum	Age	DayOff	GB
1	Male	Driving licence	Industry	Secondary education	1	3318	16589	500000	33	90+	1
2	Female	other	Education	Secondary education	0	912	4562	100000	999	0	0
3	Female	other	Finance /Banking/Insuran	Higher education	0	919	4597	100000	30	0	0
4		other	Retail	Secondary education	0	4000	20000	280000	46	0	0
5	Male	Driving licence	Industry	Secondary education	1	12170	60850	500000	42	0	0
5	Male	Driving licence	Industry	Secondary education	1	12170	60850	500000	42	0	0
7	Male	Foreign passport	Industry	Higher education	1	728	3640	100000	25	60 90	0
8	Male	Drivine licence	Finance /Banking/Insuran	Secondary education	0	2468	12341	150000	43	0	0
9	Male	Military card	Industry	Secondary education	0	825	4123	270000	36	0	0
10	Female	other	Industry	Secondary education	0	468	2340	136000	31	0	0
11	Male	Drivine licence	Industry	Secondary education	1	430	2150		45	0	0
12	Female	Driving licence	Education	Secondary education	0	2468	12341	150000	22	0	0
13	Female	Driving licence	Industry	Secondary education	0	1170	5850	480000	31	30 90	0
14	Male	Drivine licence	other	Higher education	0	1063	5317	183000	28	0	0
15	Female	other	Education	Secondary education	0	1028	5139	214000	36	0	0
16	Female	other	Education	Secondary education	0	1025	5123	15000000	43	0	0

Attribute	Field	Field Russian Name	Format/New	Format/Initial	Comment
<b>ID</b>	Identification number	Персональный номер	numeric	numerec	1234
<b>SEX</b>	Gender	Пол	numeric	char	1 - Male 2 - Female
<b>DocType</b>	Additional document	Дополнительный документ	numeric	char	1 - Military card 2 - Foreign passport 3 - Driving licence
<b>Business</b>	Employer scope of activity	Индустрия	numeric	char	1 - Finance / Banking / Insurance 2 - Industry 3 - Retail 4 - Education 5 - IT and Telecoms 6 - Other
<b>Education</b>	Education	Образование	numeric	char	1 - Graduate (Ученая степень) 2 - Higher education (Высшее) 3 - Secondary education (Среднее образование)
<b>CarStatus</b>	Ow ns car	Есть машина	numeric	char	1 - Yes 2 - No
<b>Add Income</b>	Additional income	Дополнительный доход	numeric	numeric	per month
<b>Income</b>	Income from mandatory work confirmed	доход по основному месту работы	numeric	numeric	per month
<b>CreditSum</b>	Loan amount	Сумма кредита	numeric	numerec	560000
<b>Age</b>	The age of the borrower	Возраст заемщика	numeric	numerec	43
<b>DayOff</b>	Overdue payment/days	Дни просрочки	char	char	90+/ 60_90 /30_60 / 0_30 / 0_0
<b>GB/Target</b>	Target function	Целевая функция	numeric	numeric	1- DayOff >= 90+ 0- DayOff < 90



<b>Problems of initial data set/</b> исходные («грязные») данные	<b>Format attribute/</b> формат переменной	<b>Comment/</b> комментарий
<b>1.Missing data/</b> Отсутствующие данные	Numeric/числовой, Char/текст	1. Add in (average, median, frequency...) 2. Delete this cases (rows)
<b>2.Mistakes of data/</b> Ошибки в данных	Numeric/числовой, Char/текст	1. Add in (average, median, frequency...) 2. Delete this cases (rows)
<b>3.Outliers of data /</b> Выбросы данных	Numeric/числовой	Delete this cases (rows)
<b>4.Duplicate cases(rows)</b> /Дублирующие наблюдения(строки)	Duplicate ID (observations)	Remove one of the duplicate
<b>5.Multicollinearity in the original data/</b> Мультиколлинеарность	Linear combination of variables (attributes)	Remove one of the attributes
<b>6.Digitalization of data/</b> Цифровизация данных	Numeric/числовой, Char/текст	Converting to numeric format
<b>Selection of objective function, learning and test samples/</b> Выбор целевой функции, обучающейся и тестовой выборки		
<b>Objective function/</b> Целевая функция	Binary (0,1)	“Bad” – 1; “Good” – 0.
<b>Training samples/</b> Обучающейся выборка	Sampling 70%-80%	Representative relative to the objective function (GB)/ Репрезентативная по GB
<b>Testing samples/</b>	Sampling 30%-20%	Representative relative to the objective function (GB)/ Репрезентативная по GB

# Анализ пропущенных значений («missing»)

---

- Неверные и противоречивые значения могут появиться на этапе *ввода, передача и сбора данных* в результате *опечаток, программных ограничений* (ограничение на длину переменной, ограничение размера буфера), *различных форматов записи данных*.
- **Неверные и противоречивые данные представляют проблему потому, что алгоритм логистической регрессии предполагает, что все исходные данные – корректные, и строит модель в соответствии с этим предположением, что и приводит к неверным прогнозным результатам.** При выявлении *неверных* или *противоречивых* значений, необходимо *исправить* или *удалить соответствующее наблюдение из анализа*.

# Анализ пропущенных значений

## продолжение

---

- Данные, содержащие *менее 5% пропусков*, можно считать случайными. Для данных, содержащих *от 5 до 50% пропусков*, необходимо *определить механизм их возникновения* и в соответствии с этим выбирать стратегию их заполнения. Переменные, содержащие более 50% пропущенных значения, следует удалить из анализа.
- Наиболее распространенными методами заполнения пропусков в числовых переменных являются: *заполнение константой (нулем, средним, модой, медианой, последним наблюдением)*, *заполнение из распределения*, *заполнение с помощью модели (нейросеть, дерево решений)*.
- Для обработки отсутствующих значений в категориальных переменных используется, создание *отдельной категории* для пропущенных данных, *создание бинарной переменной-индикатора*.

## Анализ выбросов («outliers»)

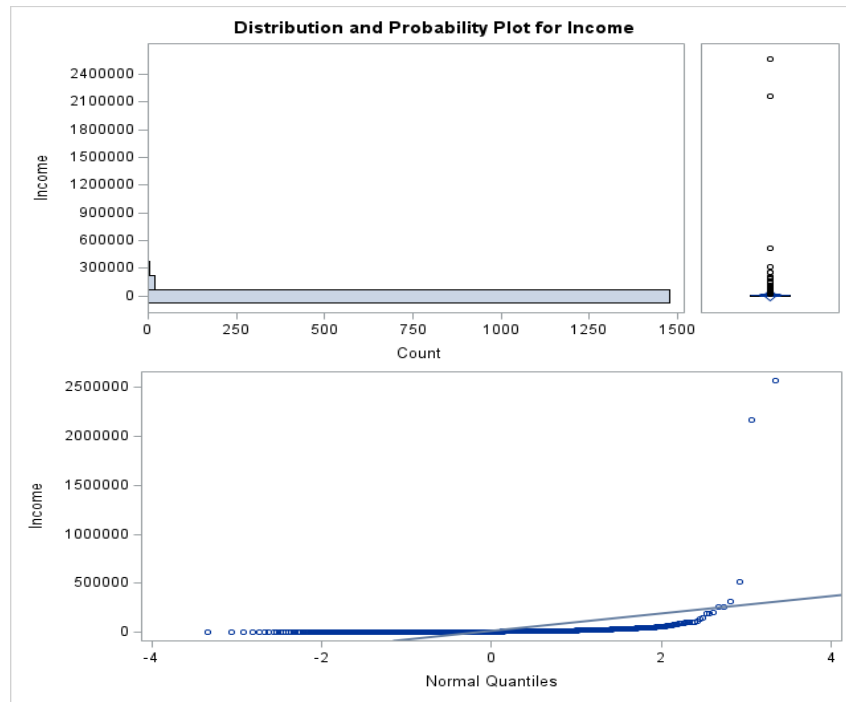
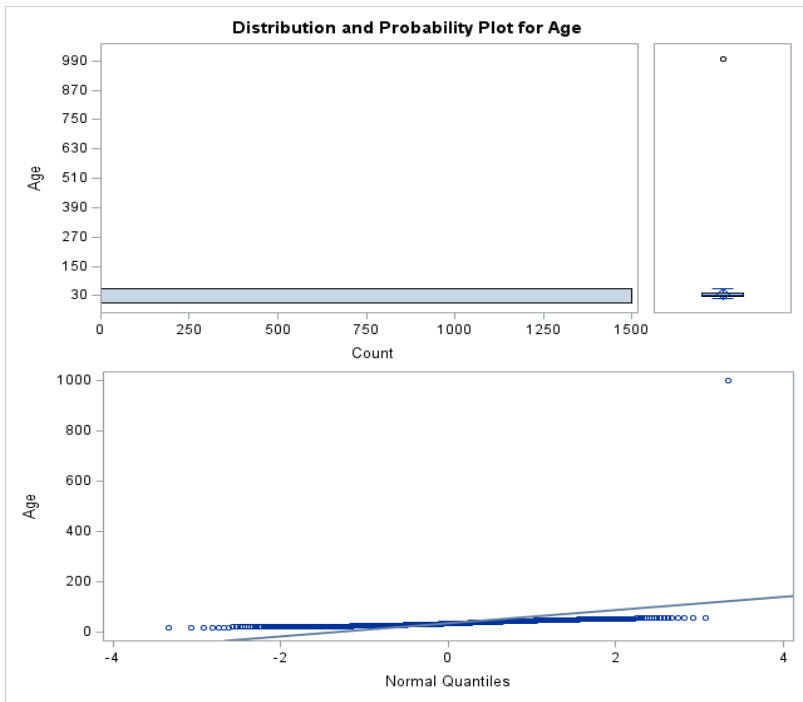
---

При небольшом количестве выбросов ( $< 5\%$ ) можно удалить их из анализа или заменить средним, или модой. При большом количестве выбросов ( $> 50\%$ ) следует выделить их в отдельную выборку для проведения анализа, поскольку это может свидетельствовать о появлении нового феномена в данных.

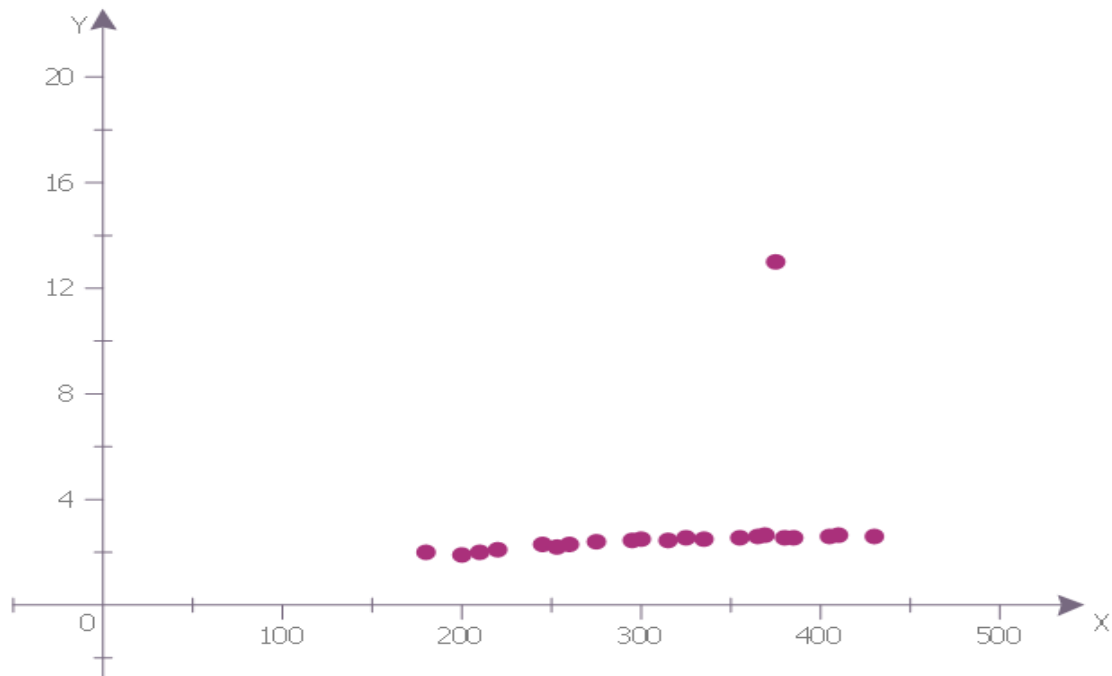
Помочь справиться с выбросами в числовой переменной может применение некоторых преобразований (min-max нормализация) и дискретизация

# Анализ выбросов («outliers»)

## Age, Income



# Визуализация выбросов



# Описание разработанной методологии

---

1. Разбиение

2. Очистение

Устранение  
дублей

Обработка  
выбросов

Коррекция  
противоречивых  
данных

Обработка  
пропусков

3. Трансформация

Приведение к  
единому  
формату

Дискретизация

Масштабирование

4. Выбор  
переменных

Мультиколлинеарность

Информативность

# 1. Разбиение данных





## 2. Очистка данных

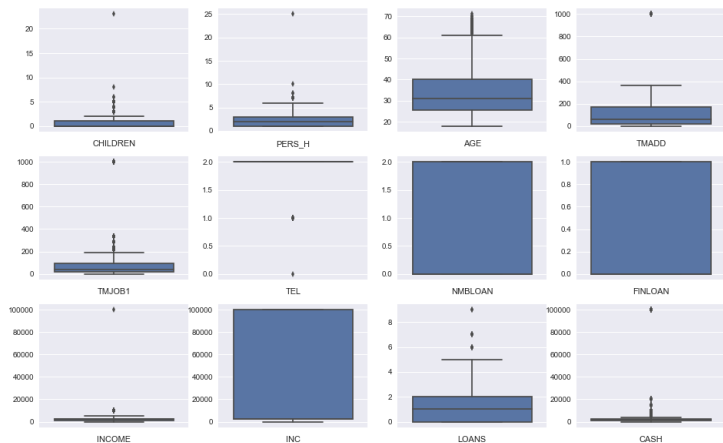


Рис. 1 – Выброс в данных

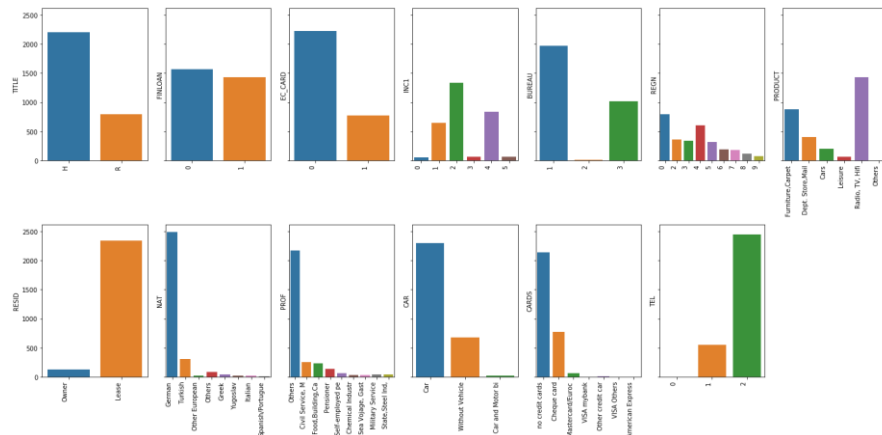


Рис. 2 – Выбросы в данных

## 3. Цифровизация данных (основные методы)

---

### Нормализация:

- ✓ Min-max нормализация.
- ✓ Нормализация стандартным отклонением.

### Форматирование:

- ✓ Приведение к единому формату.
- ✓ Дискретизация – преобразование непрерывных атрибутов в категориальные

## 4. Выбор значащих переменных

---

Мультиколлинеарность (корреляция):

- ✓ Удаление сильно коррелированных переменных.

Нормальность

- ✓ Важна при использовании параметрических методов.

Информативность

- ✓ Удаление неинформативных переменных.

# 4. Выбор значащих переменных

## Визуализация исходных данных

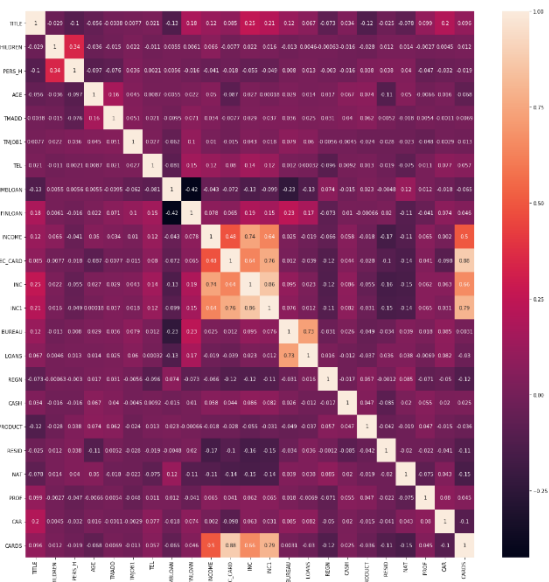


Рис. 3 – Корреляция переменных

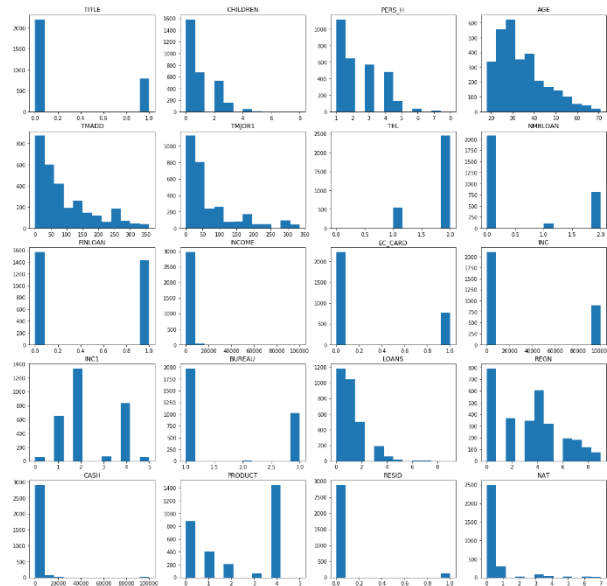


Рис. 4 – Гистограмма частот переменных

## Результаты вычислительного эксперимента по оценке точности прогнозной модели

---

	Python	SAS	SAS Enterprise Miner
Без подготовки данных («грязные»)	59%	59%	57%
С использованием предложенной методики подготовки данных	69%	73%	75%
Прирост точности прогнозной модели	+10%	+14%	+18%

## Импорт данных (Rezerv\_11.xls) в формат SAS (test1)

---

**PROC IMPORT**

DATAFILE="C:\Users\gubine\Desktop\xls\Rezerv\_11.xls"

OUT=WORK.test1

REPLACE

DBMS=XLS;

GETNAMES=YES;

**RUN;**

## Импорт данных (Rezerv\_11.xls) в формат Python 3.7.4 (test1)

---

```
def parse_excel(filepath: str) -> List[pd.core.frame.DataFrame]:  
    dfs = []  
    xl = pd.ExcelFile(filepath)  
    for sheet in xl.sheet_names:  
        dfs.append(xl.parse(sheet))  
    return dfs
```

## Импорт данных (Rezerv\_11.xls) в формат Python 3.7.4 (test1)

```
df_list = parse_excel(os.path.join(DATA_PATH, FILENAME))
main_df = df_list[0]
main_df
```

	ID	Sex	DocType	Business	Education	CarStatus	AddIncome	Income	CreditSum	Age	DayOff	GB
0	1	Male	Driving licence	Industry	Secondary education	1	3317.8	16589	500000.0	32.904110	90+	1
1	2	Female	other	Education	Secondary education	0	912.4	4562	100000.0	999.000000	0	0
2	3	Female	other	Finance /Banking/Insurance	Higher education	0	919.4	4597	100000.0	29.783562	0	0
3	4	NaN	other	Retail	Secondary education	0	4000.0	20000	280000.0	46.476712	0	0
4	5	Male	Driving licence	Industry	Secondary education	1	12170.0	60850	500000.0	41.950685	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...



# Анализ пропущенных значений (SAS) продолжение

---

```
/* create a format to group missing and nonmissing */
```

```
proc format;
```

```
value $missfmt ' ' = 'Missing' other = 'Not Missing';
```

```
value missfmt . = 'Missing' other = 'Not Missing';
```

```
run;
```

```
proc freq data=rezerv_11; /* initial data */
```

```
format _CHAR_ $missfmt.; /* apply format for the duration of this PROC */
```

```
tables _CHAR_ / missing missprint nocum noperc;
```

```
format _NUMERIC_ missfmt.;
```

```
tables _NUMERIC_ / missing missprint nocum noperc;
```

```
run;
```

# Анализ пропущенных значений

## (Python 3.7.4)

---

```
def get_missing_freq(df: pd.core.frame.DataFrame) -> pd.core.frame.DataFrame:  
    df_miss_freq = pd.DataFrame()  
    df_miss_freq['Not missing'] = df.notna().sum()  
    df_miss_freq['Missing'] = df.isna().sum()  
    return df_miss_freq
```

## Анализ дублирующих строк (SAS)

---

```
proc sort data=test1/ ** not duplicate **/  
  nodupkey out=NotDuplicate; by id;  
run;
```

Наличие одинаковых наблюдений влияет на коэффициенты регрессии, увеличивая дисперсию модели, поэтому дублирующие наблюдения должны быть найдены и удалены из анализа

## Анализ дублирующих строк (Python 3.7.4)

---

```
def print_duplicates(df: pd.core.frame.DataFrame):  
    index_list = list(df[df.duplicated()].index)  
    count = df.duplicated().sum()  
    print(f'Amount of duplicated rows: {count}\nlist of indexes: {index_list}')
```

```
def drop_duplicates(df: pd.core.frame.DataFrame) -> pd.core.frame.DataFrame:  
    return df.drop_duplicates()
```

```
print_duplicates(main_df)  
df_cleaned = drop_duplicates(main_df)
```

```
Amount of duplicated rows: 1  
list of indexes: [5]
```

```
len(df_cleaned)
```

```
1500
```

## Анализ выбросов (SAS)

```
proc univariate data=test1 robustscale plot; /* outliers */
var income age;
run;
```

<u>Income</u>				<u>Age</u>			
Extreme Observations				Extreme Observations			
Lowest		Highest		Lowest		Highest	
Value	Obs	Value	Obs	Value	Obs	Value	Obs
0	1492	263040	1310	19	1170	57.5589	1101
0	1477	312750	719	19	183	57.5808	1427
0	1471	519500	1031	20	1445	59.1041	1103
0	1465	2163580	480	20	1426	59.3973	896
0	1457	2570370	881	20	1208	999.0000	2

## Анализ выбросов (Python 3.7.4)

---

```
def print_outliers(df: pd.core.frame.DataFrame, feature: str):  
    lowest = df.sort_values([feature])[feature][:5]  
    highest = df.sort_values([feature])[feature][-5:]  
    print(f'The lowest values of {feature} feature:')  
    print(lowest)  
    print('')  
    print(f'The highest values of {feature} feature:')  
    print(highest)
```

# Влияние выбросов на прогнозные результаты

Временная шкала	Значения
1	2
2	4
3	4
4	6
5	7
6	9
7	10
8	9
9	12
10	10
11	11
12	12



**Выброс в точке 10**

# Влияние выбросов на прогнозные результаты

Временная шкала	Значения	
	1	2
	2	4
	3	4
	4	6
	5	7
	6	9
	7	10
	8	9
	9	12
	10	100
	11	11
	12	12



Выброс	36.35 (-11.97;84.67)	прогнозное значение
Нет_выброса	12.88 (10.46; 15.30)	прогнозное значение



# Влияние отсутствующих данных (missing) на прогнозные результаты

Временная шкала	Значения
1	2
2	4
3	4
4	6
5	7
6	9
7	10
8	9
9	12
10	10
11	11
12	12



Прогноз	Граница нижняя	Граница верхняя	Исх. данные
12,83	9,31	16,35	Средняя (8)
12,88	10,46	15,3	Искомая (10)

```
proc princomp data=test1 /* correlation */
    outstat=test1_stat noprint;

run;
```

<u>_TYPE_</u>	<u>_NAME_</u>	<u>ID</u>	<u>CarStatus</u>	<u>AddIncome</u>	<u>Income</u>	<u>Age</u>	<u>GB</u>
MEAN		750,9993338	0,326449034	3050,689141	15253,4457	36,15862849	0,131912059
STD		433,4468372	0,469069983	17917,40377	89587,01887	26,47480476	0,338507915
N		1501	1501	1501	1501	1501	1501
CORR	ID	1	0,017117208	-0,000380217	-0,000380217	-0,058908699	-0,003125426
CORR	CarStatus	0,017117208	1	0,007317044	0,007317044	0,010531656	-0,120234308
CORR	AddIncome	-0,000380217	0,007317044	1	1	0,018290776	-0,034779742
CORR	Income	-0,000380217	0,007317044	1	1	0,018290776	-0,034779742
CORR	Age	-0,058908699	0,010531656	0,018290776	0,018290776	1	-0,077438074
CORR	GB	-0,003125426	-0,120234308	-0,034779742	-0,034779742	-0,077438074	1

# Анализ мультиколлинеарности

## (Python 3.7.4)

---

```
stat, corr = get_stats_corr_table(main_df)
print_corr_columns(main_df)
```

```
['AddIncome', 'Income']
```

```
def get_stats_corr_table(df: pd.core.frame.DataFrame) -> Tuple[pd.core.frame.DataFrame]:
    return df.describe(), df.corr()
```

```
def print_corr_columns(df: pd.core.frame.DataFrame, corr_treshold: float = 0.9):
    df_corr = df.corr()
    corr_cols = []
    columns = list(df_corr.columns)
    for i in range(len(columns)):
        j = i + 1
        while j < len(columns):
            if df_corr[columns[i]][columns[j]] > corr_treshold:
                corr_cols.append(columns[i])
                corr_cols.append(columns[j])
            j += 1
    print(corr_cols)
```

```
def del_feature(df: pd.core.frame.DataFrame, feature: str) -> pd.core.frame.DataFrame:
    return df.drop(feature, axis=1)
```

```
/*** Digitalization Data ***/
```

```
Data test2;  
set test1;
```

```
if sex="Male" then sex_ =1;  
if sex="Female" then sex_ =2;
```

```
if DocType="Military card" then DocType_ =1;  
if DocType="Foreign passport" then DocType_ =2;  
if DocType="Drivine licence" then DocType_ =3;  
if DocType="Driving licence" then DocType_ =3;  
if DocType="other" then DocType_ =4;
```

```
if Business="Finance /Banking/Insurance" then Business_=1;
  if Business="Industry" then Business_=2;
  if Business="Retail" then Business_=3;
  if Business="Education" then Business_=4;
  if Business="IT and Telecoms" then Business_=5;
  if Business="other" then Business_=6;

  if Education="Graduate" then Education_=1;
  if Education="Higher education" then Education_=2;
  if Education="Secondary education" then Education_=3;

drop sex DocType Business Education DayOff; /* delete Char variables */

run;
```

```
def digitalize(df: pd.core.frame.DataFrame) -> pd.core.frame.DataFrame:
    df_new = df.copy()
    obj_columns = df_new.select_dtypes(include=['object']).columns
    for feat in obj_columns:
        df_new[feat] = pd.factorize(df_new[feat])[0]
    return df_new
```

# Цифровизация данных

## (Python 3.7.4)

```
digitalize(df_cleaned_2)
```

	ID	Sex	DocType	Business	Education	CarStatus	Income	CreditSum	Age	DayOff	GB
0	1	0	0	0	0	1	16589	500000.0	32.904110	0	1
1	2	1	1	1	0	0	4562	100000.0	999.000000	1	0
2	3	1	1	2	1	0	4597	100000.0	29.783562	1	0
3	4	-1	1	3	0	0	20000	280000.0	46.476712	1	0
4	5	0	0	0	0	1	60850	500000.0	41.950685	1	0
...	...	...	...	...	...	...	...	...	...	...	...
1496	1497	0	3	3	2	1	146520	350000.0	56.287671	1	0
1497	1498	0	3	0	0	0	7891	320000.0	31.956164	1	0
1498	1499	1	1	0	1	1	42600	500000.0	39.928767	1	0
1499	1500	1	0	5	2	1	9832	250000.0	29.945205	1	0
1500	1501	0	3	0	1	1	21511	400000.0	30.109589	1	0

1500 rows × 11 columns

# Цифровизация данных

## (промежуточная)

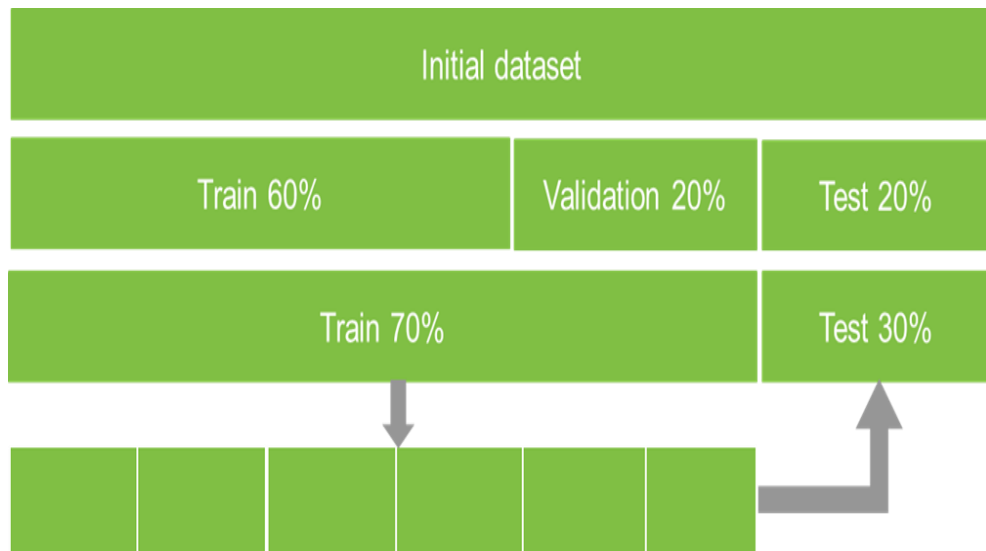
ID	CarStatus	AddIncome	Income	CreditSum	Age	GB	sex_	DocType_	Business_	Education_
1	1	3317,8	16589	500000	32,9	1	1	3	2	3
2	0	912,4	4562	100000	999	0	2	4	4	3
3	0	919,4	4597	100000	29,8	0	2	4	1	2
4	0	4000	20000	280000	46,5	0		4	3	3
5	1	12170	60850	500000	42	0	1	3	2	3
5	1	12170	60850	500000	42	0	1	3	2	3
7	1	728	3640	100000	25,4	0	1	2	2	2
8	0	2468,2	12341	150000	43,1	0	1	3	1	3
9	0	824,6	4123	270000	36,3	0	1	1	2	3
10	0	468	2340	136000	30,5	0	2	4	2	3
11	1	430	2150		44,7	0	1	3	2	3
12	0	2468,2	12341	150000	21,9	0	2	3	4	3
13	0	1170	5850	480000	30,8	0	2	3	2	3
14	0	1063,4	5317	183000	27,9	0	1	3	6	2
15	0	1027,8	5139	214000	36	0	2	4	4	3
16	0	1024,6	5123	15000000	43,1	0	2	4	4	3



ID	CarStatus	Income	CreditSum	Age	GB	sex_	DocType_	Business_	Education_
1	1	16589	500000	32,9	1	1	3	2	3
2	0	4562	100000	35,2	0	2	4	4	3
3	0	4597	100000	29,8	0	2	4	1	2
4	0	20000	280000	46,5	0	1	4	3	3
5	1	60850	500000	42	0	1	3	2	3
7	1	3640	100000	25,4	0	1	2	2	2
8	0	12341	150000	43,1	0	1	3	1	3
9	0	4123	270000	36,3	0	1	1	2	3
10	0	2340	136000	30,5	0	2	4	2	3
11	1	2150	291721	44,7	0	1	3	2	3
12	0	12341	150000	21,9	0	2	3	4	3
13	0	5850	480000	30,8	0	2	3	2	3
14	0	5317	183000	27,9	0	1	3	6	2
15	0	5139	214000	36	0	2	4	4	3
17	0	15429	200000	31	1	1	4	4	2

# Подготовка данных для анализа

## Data partition



## Data cleaning

- Removing duplicates
- Outliers
- Inconsistent data
- Missing data

# Выбор тренировочной и тестовой выборок (SAS)

---

```
/*The program divides file (ONE) into two (Train and Test)  
randomly*/
```

```
DATA One;  
  SET test1;  
  LABEL x = 'Random number';  
  x=RANUNI(int(time()));  
RUN;
```

There were 1499 observations read from the data set WORK.ONE.  
The data set **WORK.ONE** has 1499 observations and 11 variables.

```
DATA Test;  
  SET One;  
  IF x>0.75;  
RUN;
```

The data set **WORK.TEST** has 366 observations and 11 variables.

```
DATA Train;  
  SET One;  
  IF x<=0.75;  
RUN;
```

The data set **WORK.TRAIN** has 1133 observations and 11 variables.

# Выбор тренировочной и тестовой выборки (Python 3.7.4)

```
def get_train_test_dfs(df: pd.core.frame.DataFrame, target_col: str) -> Tuple[np.ndarray]:  
    df = df.copy()  
    target = df[target_col].values  
    df.drop(target_col, axis=1, inplace=True)  
    X_train, X_test, y_train, y_test = train_test_split(  
        df.values, target, test_size=0.2558, random_state=42)  
    return X_train, X_test, y_train, y_test
```

```
X_train, X_test, y_train, y_test = get_train_test_dfs(df_cleaned_2, 'GB')  
print(f'X_train shape: {X_train.shape}')  
print(f'X_test shape: {X_test.shape}')  
print(f'y_train shape: {y_train.shape}')  
print(f'y_test shape: {y_test.shape}')
```

```
X_train shape: (1116, 10)  
X_test shape: (384, 10)  
y_train shape: (1116,)  
y_test shape: (384,)
```

## Анкетные данные заемщиков

Attribute	Field	Field Russian Name	Format/New	Format/Initial	Comment
ID	Identification number	Персональный номер	numeric	numerec	1234
SEX	Gender	Пол	numeric	char	1 - Male 2 - Female
DocType	Additional document	Дополнительный документ	numeric	char	1 - Military card 2 - Foreign passport 3 - Driving licence
Business	Employer scope of activity	Индустрия	numeric	char	1 - Finance / Banking / Insurance 2 - Industry 3 - Retail 4 - Education 5 - IT and Telecoms 6 - Other
Education	Education	Образование	numeric	char	1 - Graduate (Ученая степень) 2 - Higher education (Высшее) 3 - Secondary education (Среднее образование)
CarStatus	Owens car	Есть машина	numeric	char	1 - Yes 2 - No
AddIncome	Additional income	Дополнительный доход	numeric	numeric	per month
Income	Income from mandatory work confirmed	доход по основному месту работы	numeric	numeric	per month
CreditSum	Loan amount	Сумма кредита	numeric	numerec	560000
Age	The age of the borrower	Возраст заемщика	numeric	numerec	43
DayOff	Overdue payment/days	Дни просрочки	char	char	90+ / 60_90 / 30_60 / 0_30 / 0_0
GB/Target	Target function	Целевая функция	numeric	numeric	1 - DayOff >= 90+ 0 - DayOff < 90

Модель логистической регрессии

$$\ln\left(\frac{p_i}{1-p_i}\right) = b_0 + b_1x_{1,1} + \dots + b_kx_{i,j} + \varepsilon_i$$

где  $p_i$  — вероятность наступления дефолта по кредиту для  $i$ -го заемщика;

$x_{ij}$  — значение  $j$ -ой независимой переменной;

$b_0$  — независимая константа модели,

$b_j$  — параметры модели;

$\varepsilon_i$  — компонент случайной ошибки.

## Экспорт преобразованных данных из формата SAS (test1) в rezerv1.xls.

---

```
proc export data=test1  
  outfile="C:\Users\gubine\Desktop\xls\rezerv1.xlsx"  
  dbms=xlsx  
  replace;  
run;
```

## Анкетные данные заемщиков

Attribute	Field	Field Russian Name	Format/New	Format/Initial	Comment
<b>ID</b>	Identification number	Персональный номер	numeric	numerec	1234
<b>SEX</b>	Gender	Пол	numeric	char	1 - Male 2 - Female
<b>DocType</b>	Additional document	Дополнительный документ	numeric	char	1 - Military card 2 - Foreign passport 3 - Driving licence
<b>Business</b>	Employer scope of activity	Индустрия	numeric	char	1 - Finance / Banking / Insurance 2 - Industry 3 - Retail 4 - Education 5 - IT and Telecoms 6 - Other
<b>Education</b>	Education	Образование	numeric	char	1 - Graduate (Ученая степень) 2 - Higher education (Высшее) 3 - Secondary education (Среднее образование)
<b>CarStatus</b>	Owens car	Есть машина	numeric	char	1 - Yes 2 - No
<b>Add Income</b>	Additional income	Дополнительный доход	numeric	numeric	per month
<b>Income</b>	Income from mandatory work confirmed	доход по основному месту работы	numeric	numeric	per month
<b>CreditSum</b>	Loan amount	Сумма кредита	numeric	numerec	560000
<b>Age</b>	The age of the borrower	Возраст заемщика	numeric	numerec	43
<b>DayOff</b>	Overdue payment/days	Дни просрочки	char	char	90+ / 60_90 / 30_60 / 0_30 / 0_0
<b>GB/Target</b>	Target function	Целевая функция	numeric	numeric	1 - DayOff >= 90+ 0 - DayOff < 90

Классическая формула линейной регрессионной модели имеет следующий

$$y = a + b_1 x_1 + b_2 x_2 + \dots + b_n x_n$$

Здесь зависимая переменная является линейной функцией независимых переменных. Делая логит-преобразование

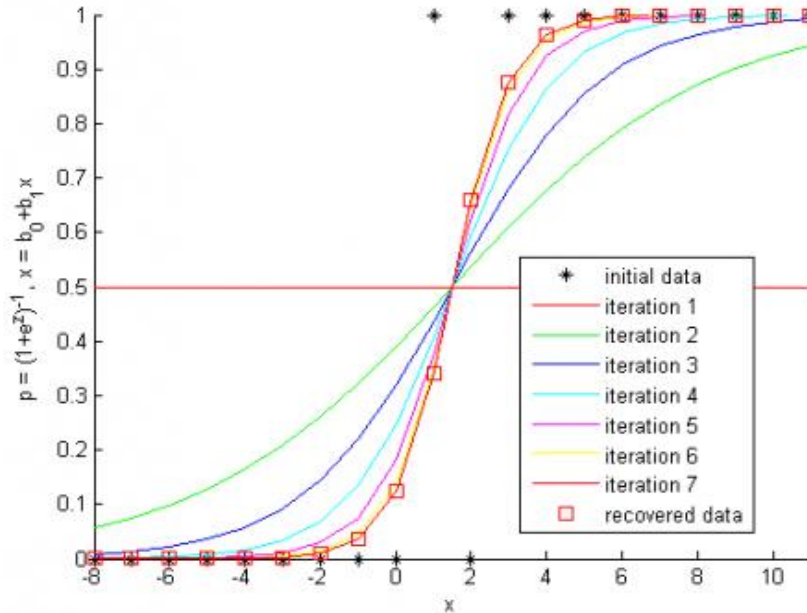
$$p = \frac{1}{1 + e^{-y}}$$

получим

$$\ln\left(\frac{p_i}{1 - p_i}\right) = b_0 + b_1 x_{i,1} + \dots + b_k x_{i,j} + \varepsilon_i$$

# Итерационное вычисление параметров логистической регрессии

Таблица: Анкетные данные заемщиков



Модель логистической регрессии

$$\ln\left(\frac{p_i}{1-p_i}\right) = b_0 + b_1 x_{i,1} + \dots + b_k x_{i,j} + \varepsilon_i$$

где  $p_i$  — вероятность наступления дефолта по кредиту для  $i$ -го заемщика;

$x_{ij}$  — значение  $j$ -ой независимой переменной;

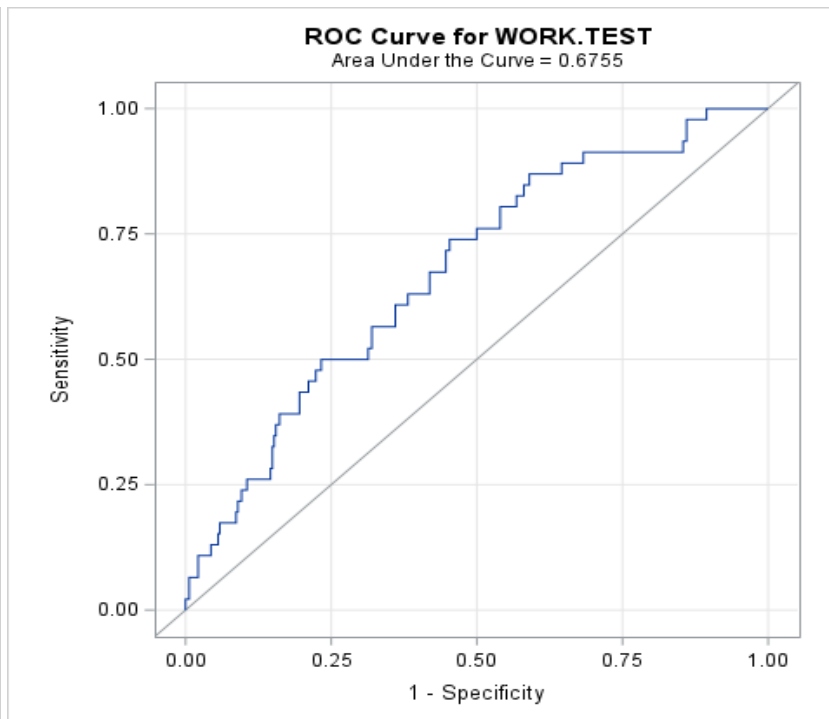
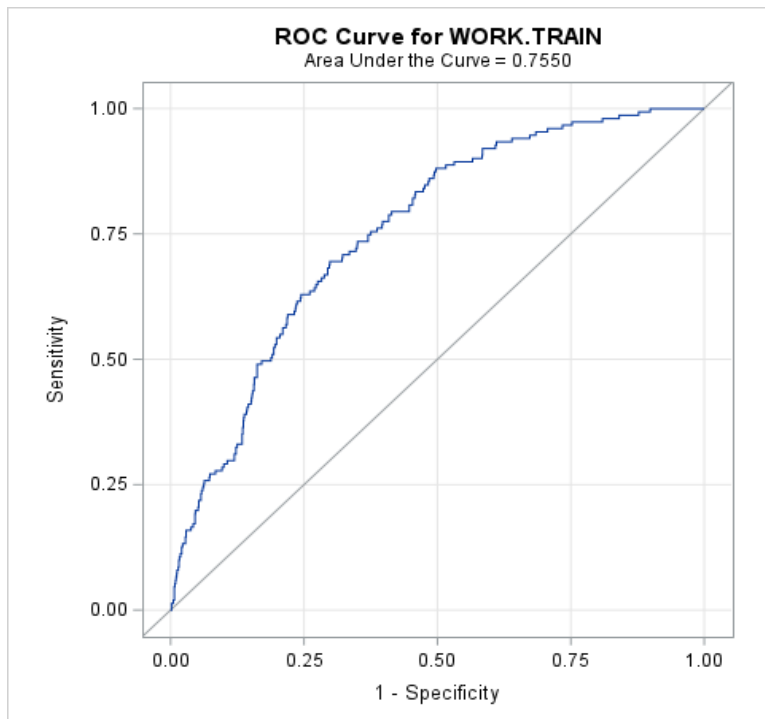
$b_0$  — независимая константа модели,

$b_j$  — параметры модели;

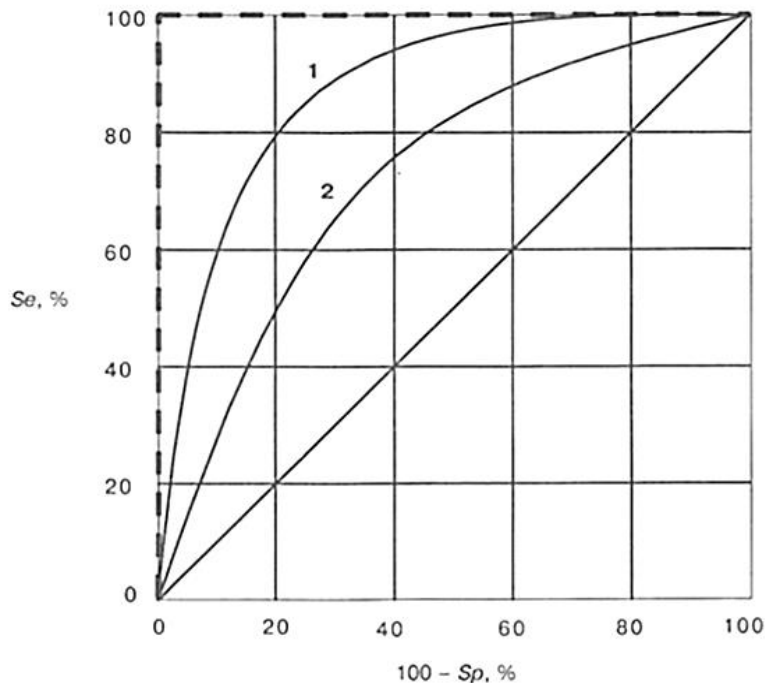
$\varepsilon_i$  — компонент случайной ошибки.



# Результаты классификации для тренировочной и тестовой выборок

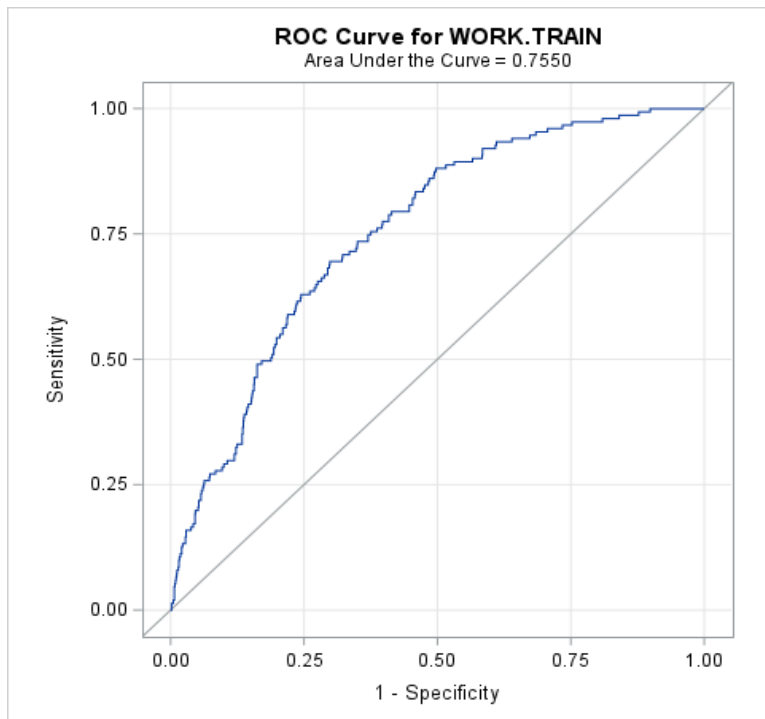


# Итерационное вычисление параметров логистической регрессии



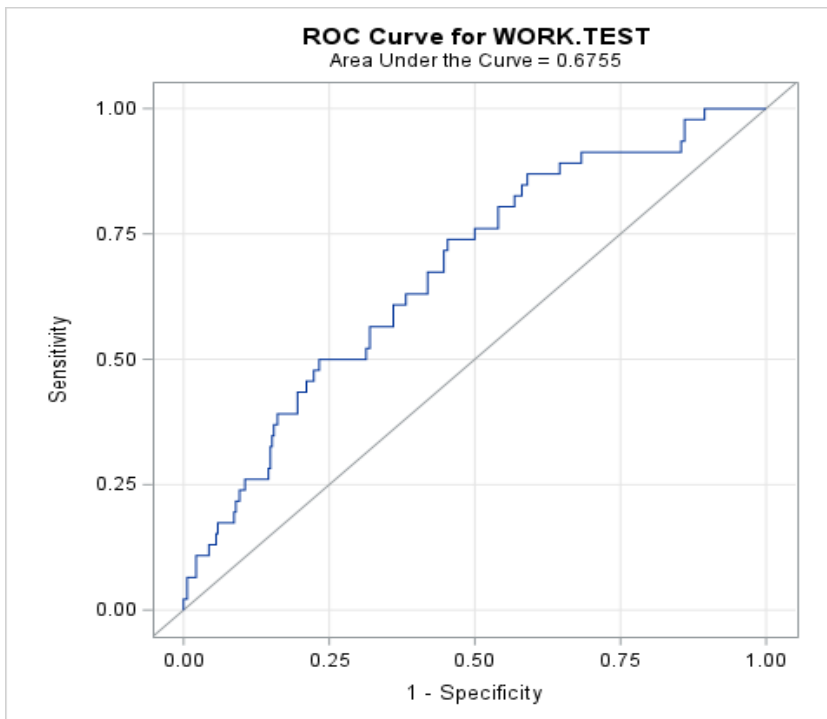
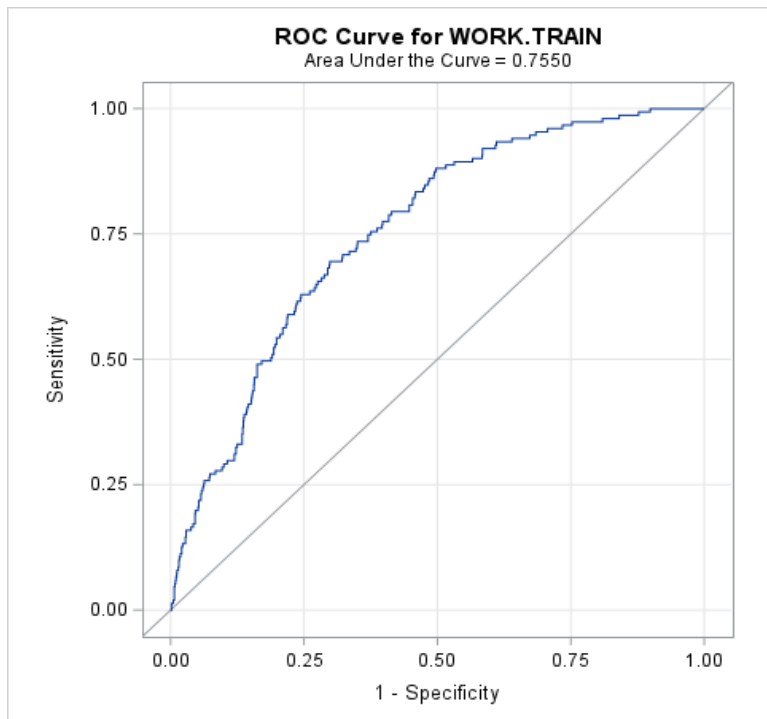
**ROC-кривая** (receiver operating characteristics curve) – позволяет рассмотреть все пороговые значения для данного классификатора. Идеальная ROC-кривая проходит через левый верхний угол, соответствуя классификатору, который дает высокое значение полноты при низкой доле ложно положительных примеров. Точка, ближе всего расположенная к верхнему левому углу дает ROC близкой к 1, что говорит о высокой силе предсказательной модели и ROC близкой к 0.5 говорит о слабой силе предсказательной модели, равносильной равновероятной.

# ROC кривая логистической регрессии



		Истинные значения	
		0	1
Предсказанные	0	True Negative (TN)	False Negative (FN)
	1	False Positive (FP)	True Positive (TP)

# Итерационное вычисление параметров логистической регрессии



## Скоринговая карта (фрагмент)

		Group	Scorecard Points	Weights of Evidence	Event Rate	Percentage of Population	Coefficient
AGE	low <= AGE < 29	1.00	-10	-0.47	7.33	13.88	-1.63
	29 <= AGE < 33	2.00	1	-0.22	5.79	17.58	-1.63
	33 <= AGE < 37	3.00	22	0.22	3.81	19.43	-1.63
	37 <= AGE < 41	4.00	23	0.25	3.70	17.48	-1.63
	41 <= AGE < 44	5.00	31	0.40	3.20	11.56	-1.63
	44 <= AGE < 48	6.00	27	0.34	3.42	10.82	-1.63
	48 <= AGE < high	7.00	0	-0.25	6.00	9.25	-1.63
Car	Car = 1	1.00	8	-0.33	6.44	37.37	-0.41
	Car = 0, 2	2.00	15	0.26	3.69	62.63	-0.41
Children	Children = 1	1.00	8	-0.13	5.36	67.35	-0.90

**В данной работе** предложена методика подготовки данных для построения прогнозных моделей классификации. Этапы подготовки данных включают в себя следующие этапы: **1.** проверку исходных данных на ошибки («описки», ошибки в форматах), **2.** на отсутствие данных (“missing”), **3.** на выбросы данных (“outliers”), **4.** на наличие дублирующих строк (наблюдений), **5.** на проверку исходных объясняющих переменных (атрибутов) на мультиколлинеарность и **6.** трансформация исходных данных в цифровой формат («цифровизация») и **7.** выбор целевой переменной.

Полученная методика реализована в программных пакетах Python, SAS, SAS Enterprise Miner.

Сравнение точности результатов, полученных без подготовки данных и с применением предложенной методики подготовки данных показало повышение предсказательной силы прогнозной модели почти на 20%.

Наибольшую точность (75%) демонстрирует решение, полученное с помощью SAS Enterprise Miner.

1. Губин Е.И. Методика подготовки больших данных для прогнозного анализа. «Наука и бизнес: пути развития». Выпуск № 3(105). 2020, 2020. – [С. 33-35].
2. Губин Е.И. Методология подготовки больших данных для прогнозного анализа. Современные технологии, экономика и образование: Сборник трудов Всероссийской научно-методической конференции. / Томский политехнический университет. – Томск: Изд-во Томского политехнического университета, 2019. – 139с. – [С. 25-28].
3. Вершинин А.С., Губин Е.И. Применение инструмента DATA MINING для оценки кредитоспособности заемщика // Информационные технологии в науке, управлении, социальной сфере и медицине: Труды V Междунар. конференции. – Томск, 2018. – Т.2. – С. 18-21.
4. Вершинин А.С., Губин Е.И. Использование инструментов SAS для оценки рисков заемщиков // Молодежь и современные информационные технологии: Труды XVI Междунар. научно - практической конференции студентов, аспирантов и молодых ученых. Томск, 2018г. - С. 379-380.
5. Руководство по кредитному скорингу /под ред. Элизабет Мэйз; пер. с англ. И.М.Вороненко. - Минск: Гревцов Паблицер, 2008. - 464с.

# Спасибо за внимание И...

.. будущую подготовку данных для анализа  
по предложенной методике